



[COM Home](#) | [Events and Training](#) | [COM Support](#) | [MSDN Online](#) | [Developer Products](#)

⊕ [News](#)

→ [White Papers](#)
[Presentations](#)

[Case Studies](#)

⊕ [Technologies](#)

⊕ [Resources](#)

White Papers

[COM Papers](#) [DCOM Papers](#) [COM+ Papers](#) [MTS Papers](#) [ActiveX Papers](#)

The Component Object Model (COM) and its related COM-based technologies of DCOM, COM+, MTS and ActiveX® comprise the most widely-used component software model in the world. As a result, several technical white papers, FAQs, and other documentation have been written on these technologies.

For a comprehensive collection of technical articles and white papers on COM and COM-based technologies, visit the [MSDN Library](#). ➡ In the Library Table of Contents, expand the "Technical Articles" heading, then expand the "Component Object Model" heading. For broader, more general information on COM-based technologies, expand the "Backgrounders" heading, then expand the "Component Object Model" heading.

COM Papers

[Middle-Tier High-Speed Data Caching Involving COM/MTS and COM+](#) ➡

This article provides information and resources for Microsoft technologies that can be used for middle-tier caching with data accessed from a COM component hosted in MTS or COM+.

[Non-Blocking Method Calls](#) ➡

With the introduction of non-blocking method calls, Microsoft has placed a powerful new tool in the hands of COM developers. Clients can use non-blocking method calls to exploit parallelism without the pain of multi-threading, and servers can handle calls asynchronously for vastly improved scalability. This paper provides a thorough introduction and shows the many interesting applications of non-blocking method calls.

[Creating a Microsoft Office 2000 COM Add-in](#) ➡

This paper describes how to use Microsoft® Office 2000 Developer to develop a COM component that creates a simple report in Microsoft FrontPage®, Microsoft Word, or Microsoft PowerPoint® with data from a Microsoft Access database.

[COM Add-ins Part I: Introducing an Office 2000 Solution for the Entire \(Office\) Family](#) ➡

This paper helps you get started with COM add-ins in Office 2000 by taking a look at the architecture of the COM add-in technology. The compatibility of COM add-ins across all Microsoft Office products

means you can write your add-in once, and then use the add-in in any Office product. Also see "[COM Add-ins Part II: Building a COM Add-in for Outlook 2000](#)" ➡ which puts the theory into practice by building a COM add-in for Outlook 2000.

Building COM Property Pages with the Active Template Library ➡

This paper discusses how to build COM property pages with the Active Template Library (ATL) and includes a demonstration on how to set some of the properties of a text document within the Microsoft® Visual C++® editor.

COM Internet Services ➡

COM Internet Services (CIS) introduces support for a new Distributed COM (DCOM) transport protocol known as Tunneling Transmission Control Protocol (TCP) that allows DCOM to operate over TCP port 80. This allows a client and a server to communicate in the presence of most proxy servers and firewalls, thereby enabling a new class of COM-based Internet scenarios.

Developing COM Add-Ins for Microsoft Office 2000 ➡

Because of the new extensibility interface in Office 2000, developers can now develop solutions for Office 2000 with any language that can create COM components. COM provides a unified development environment and add-in model enabling easier application extension development for Office 2000.

Integrating Java and COM ➡

Because the Microsoft virtual machine provides automatic mapping that allows any Java object to be a COM object and any COM object to be accessible as Java object, Java is an ideal language for taking advantage of the Microsoft Component Object Model (COM). Learn how Java/COM integration allows developers to exploit the benefits of Java and the power of the Windows® operating system in this white paper.

Microsoft Component Services - a Technical Overview

Microsoft's component services, Component Object Model (COM), Distributed COM (DCOM), Microsoft Transaction Server (MTS), Microsoft Internet Information Server (IIS), and Microsoft Message Queue (MSMQ), focus on building better enterprise applications. This technical overview explains how Microsoft component services provide a powerful, easy-to-use, and flexible platform for building distributed applications.

COMponents ➡

The key to building interchangeable plug-ins lies in knowing how to create Component Object Model (COM) objects and interfaces. Find out how to develop a series of interchangeable COM plug-ins in this article.

From CPP to COM ➡

Using an existing project written in C++, this white paper turns it into a component-based design that runs the object in a separate

process, leveraging the COM infrastructure provided natively on all new Microsoft operating systems. It explains how and why features are implemented in COM as a means of providing a robust, portable, and transparent model for distributed component software.

[FAQ: COM Security Frequently Asked Questions](#) ➔

Secure distributed applications can be created using the COM/DCOM security model. Developers of secure COM components can find tips and techniques, as well as troubleshooting information, in this document.

[COM: Technical Overview](#) ➔

COM is an open software architecture for the cross-platform development of client/server applications based on object-oriented technology. This white paper provides a comprehensive technical overview of COM.

[ObjectWatch Newsletter, Number 7: Java - The Microsoft Perspective, MS](#) ➔ Roger Sessions

In this article, Roger Sessions examines Microsoft's Java perspective.

 [Back to top](#)

DCOM Papers

[Using Distributed COM with Firewalls](#), Michael Nelson

With the continued growth of both the Internet and the Web, the occurrence of firewalls also increases; fortunately, working with DCOM and firewalls is easy. In this paper, Michael Nelson provides simple step-by-step information on how to configure your environment to provide secure DCOM functionality for your Internet applications. Nelson covers controlling the range of available Transmission Control Protocol (TCP) ports, configuring for various network protocols, and much more.

[DCOM Architecture](#) ➔, Microsoft Corporation

Focusing on the interworkings of DCOM, this paper targets the application developer who wants to create state-of-the-art applications that scale equally well to the Intranet, the Internet, and beyond.

[DCOM Technical Overview](#) ➔, Microsoft Corporation

Providing a high-level overview of DCOM, this white paper explores how you can use DCOM to solve the hardest problems associated with distributed applications.

[DCOM Business Overview](#) ➔, Microsoft Corporation

DCOM is a technology that enables software components to communicate directly with each other across networks, including the Internet and intranets. This business overview explains how DCOM can make your development and deployment process more efficient.

Microsoft Component Services - a Technical Overview

Microsoft's component services, Component Object Model (COM), Distributed COM (DCOM), Microsoft Transaction Server (MTS), Microsoft Internet Information Server (IIS), and Microsoft Message Queue (MSMQ), focus on building better enterprise applications. This technical overview explains how Microsoft component services provide a powerful, easy-to-use, and flexible platform for building distributed applications.

Distributed Component Object Model (DCOM) Binary Protocol

Specifications, papers, links to Web sites, and recommended readings provide the necessary resources for understanding DCOM—a protocol that enables software components to communicate directly over a network in a reliable, secure, and efficient manner.

 [Back to top](#)


COM+ Papers

Object-Oriented Software Development Made Simple with COM+ Run-Time Services ➡

Here you will find MSJ's introduction from the first article in a four-part series on COM+. In this article, you will receive an overview of object-oriented development issues and a discussion of how the introduction of COM+ will resolve many of them. Future articles will delve into the COM+ programming model and the services provided by COM+. This series is written by Mary Kirtland, a Program Manager on the Microsoft COM team.

The COM+ Programming Model Makes It Easy to Write Components in Any Language ➡

This is the entire text of Mary Kirtland's second article on COM+. In this article, you will learn how you can use COM+ to write components in any language.

 [Back to top](#)

MTS Papers

Duamish Books, Phase 3.5 ➡

Find out how MSDN used Microsoft Transaction Server (MTS) to distribute components and achieve a physical three-tier architecture with their Duamish Books sample application. The site enables you to download the sample and source code and provides several articles that describe many issues faced with the MTS implementation. For an introduction to the Duamish Books sample and links to previous phases of development, see [An Introduction to the Duamish Books Sample](#). ➡

Comparing Microsoft Transaction Server to Enterprise JavaBeans

This paper provides a comparison between Microsoft Transaction Server (MTS) and Enterprise JavaBeans, the specification driven by Sun Microsystems Inc. The two models are compared in terms of transactions support, controlling complexity, portability, interoperability, language choice and administration, among other topics.

Microsoft Transaction Server 2.0 Review Guide

MTS simplifies the development and deployment of server-centric applications built using Microsoft COM-based technologies. As such, MTS is ideal for developing line-of-business and electronic commerce applications with Web-based interfaces, leveraging powerful development tools such as Visual Basic for scalable, three-tiered development, and building production-quality applications with distributed object technologies. This guide provides the information you will need to make decisions about *when* and *how* to use MTS.

Microsoft Transaction Server 2.0 Market Bulletin

This bulletin describes the functionality provided in Microsoft Transaction Server (MTS) 2.0 as well as availability and licensing terms.

MTS Databases and Transactions FAQ, version 1.8 ➡

Covers issues concerning MTS transactions and database integration.

How Microsoft Transaction Server Changes the COM Programming Model ➡, David Chappell and The Third Wave ➡, Don Box

January's *Microsoft Systems Journal (MSJ)* contains MTS articles from two COM gurus. David Chappell discusses how MTS lets you write simple COM-based servers that are still powerful and scalable. Don Box points out how MTS extends the traditional object-oriented paradigm.

Microsoft Component Services - a Technical Overview

Microsoft's component services, Component Object Model (COM), Distributed COM (DCOM), Microsoft Transaction Server (MTS), Microsoft Internet Information Server (IIS), and Microsoft Message Queue (MSMQ), focus on building better enterprise applications. This technical overview explains how Microsoft component services provide a powerful, easy-to-use, and flexible platform for building distributed applications.

The Microsoft Transaction Server (MTS): Transactions Meet Components, David Chappell, Patricia Seybold Group

Describes how to use MTS and MTS components to write transactional applications based on COM and DCOM.

Microsoft Transaction Server: Guidelines for Development Tool Integration ➡ (127.5K) Using Microsoft's anonymous FTP from

anonymous@ftp.microsoft.com, download the "Microsoft Transaction Server: Guidelines for Development Tool Integration" paper. This paper presents requirements and suggestions that are designed to

help you integrate your application development tools with MTS.

Microsoft Distributed Transaction Coordinator: Resource Manager Implementation Guide, Version 6.5 ➡

Find out how to implement an OLE Transaction-compliant resource manager by downloading "Microsoft Distributed Transaction Coordinator: Resource Manager Implementation Guide, Version 6.5." Resource managers that support the OLE Transaction interfaces are capable of participating in transactions coordinated by the MS DTC. Log on as user anonymous to download the document. This document is intended for system software developers who implement transaction-protected resource managers, including relational database systems, object-oriented database systems, file systems, document storage systems, and message queuing systems.

Developing OLE Transaction Resource Managers: Readme ➡ (1.8K)

Phil Garrett, MS DTC Program Manager, provides sources of information as well as guidelines for anyone who is interested in developing OLE Transaction compliant resource managers. His "Developing OLE Transaction Resource Managers: Readme file" is available as an FTP download. Log on as user anonymous to download the Readme file.

Internet-Draft: Transaction Internet Protocol ➡ (42.6K)

The Transaction Internet Protocol (TIP) paper, in Internet-Draft form, is available as an FTP download for users signed in as anonymous on the Microsoft FTP site. The "Internet-Draft: Transaction Internet Protocol" proposes a simple, easily implemented, protocol to guarantee that the different nodes which cooperate on some work in applications fulfill that work automatically.

Using Microsoft Transaction Server 2.0 with COM Transaction Integrator 1.0 ➡

This technical white paper describes how the Microsoft® COM Transaction Integrator (COMTI) 1.0 extends Microsoft Transaction Server (MTS) 2.0 into the mainframe transaction processing world.

Enlisting Resources in Transactions with MTS ➡

Wondering which protocols are supported when enlisting resources in MTS transactions? Find out from this white paper.

Automating MTS Client Installation ➡

This technical white paper describes how you can use the client install utility to automatically generate installation executables for clients of MTS applications.

Automating MTS Deployment ➡

The use of scriptable administration objects to automate the deployment and distribution of MTS packages is explained in this technical white paper.

Business Logic in MTS Components ➡

This MTS white paper describes the difference between writing and

using fine-grained and coarse-grained components.

Database Access Interfaces with MTS ➡

The database access interface options for MTS applications are explained in this technical white paper.

MTS Error Diagnosis ➡

Discover how to determine the source of an error in your MTS version 1.0 application by perusing this technical white paper.

DLLs, Type Libraries, and MTS ➡

This technical white paper describes how an MTS component is a type of COM component that executes in the MTS run-time environment, and therefore must satisfy COM and DLL requirements.

Packaging MTS Components ➡

The design issues associated with defining package boundaries for MTS components are explained in this white paper.

Parameter Passing in MTS ➡

Parameter types and marshaling, objects as parameters, passing MTS object references, callbacks, and considerations when returning large amounts of data are all covered in this comprehensive MTS white paper.

Holding State in Objects with MTS ➡

The guidelines in deciding where state is held in your application as well as where holding state is desirable is detailed in this paper.

Transaction Context Objects in MTS ➡

An overview of transaction context objects with examples is provided in this white paper.

Determining Transaction Outcome in MTS ➡

This white paper discusses how applications determine the outcome of a transaction; that is, whether the transaction will commit or abort.

Calling MTS Components ➡

This white paper summarizes rules for calling components, including registry and security settings required by DCOM, when to use Remote Automation, calling through HTTP, creating and calling components from an Active Server Page, and calling components from a browser.


 [Back to top](#)

ActiveX Papers

How to Write and Use ActiveX Controls for Microsoft Windows CE ➡

This paper introduces ActiveX controls, discusses how to build and

distribute ActiveX controls for Windows CE, and describes how to use ActiveX controls in Windows CE-based applications.

 [Back to top](#)

Last Updated: 1/9/02

©2002 Microsoft Corporation. All rights reserved. [Terms of Use](#).